Modelling and Implementation of an Artificial Economy through JaCaMo

Igli Hakrama

Computer Engineering Department
Epoka University
Tirana, Albania

Neki Frashëri

Center for R&D in ICT
Polytechnic University of Tirana
Tirana, Albania

ABSTRACT
This paper proposes and implements a new approach to simulate an artificial economy. Based on the needs of economists to study the economy, the paper focuses on agent-based methods. First, a literature review of these methods is given and different research requirements are outlined. Then a new proposal is given through the comparison of agent-based modeling techniques and those used in multi-agent systems, from whom the JaCaMo framework was chosen as the implementation platform. The paper continues with the conceptual model of the artificial economy. A description of the economic model is explained in detail and then a detailed analysis is considered to implement the model. The implementation of this simulator is presented and the techniques used to make it are explained in detail. The paper gives some important notes over the methodology used in this implementation and makes some recommendations for future research work

Keywords
multi-agent systems, agent-based modeling, artificial economy, multi-agent oriented programming.

1. INTRODUCTION
The numerous financial crises of the past decade have given greater impetus to researchers in looking to find new approaches to study the economy and prevent future crises. Traditional economic tools appear inadequate in explaining many of the dynamics of economic systems. Therefore, new approaches are needed to improve economic understanding and solve future challenges [1]. An artificial economy is a computer simulation that can be used to address such problems and better understand socio-economical processes [2]. The use of simulations gives researchers a tool that they may use to gain a richer understanding in examining a problem. These simulations try to explain the world by mimicking the behavior of real beings or systems [3].

Artificial economy tries to understand the economic dynamics by defining multiple agents with different behaviors in a simulation environment. Such systems are usually called complex adaptive systems (CAS) [4] and are usually dealt with agent-based modelling (ABM). ABM can be considered as a new kind of research which makes a connection between micro and macro-world by developing models based on smaller units called agents. It is not considered as a competitor of other research methods, but as a very useful complementary research method to build better and realistic models of the real world [5]. In artificial economy, ABM are also known as agent-based computational economics (ACE) and different researchers are trying to fulfil some of its requirements based on agent types design [6].

With the development of many techniques in computer science and especially in multi-agent systems (MAS), there

exist a need to make better and more useful ABM tools to fulfil the requirements of artificial economy. This work presents a new MAS solution to study economy. The next section introduces the state of the art related to the main concepts and works in artificial economy, then it continues with the proposed solution based on literature review and previous research studies. On the following section, it is given the conceptual model of the artificial economy used in this work, where we describe the model and give a more detailed analysis on agent interactions. The implementation section provides a detailed explanation about the implementation of the artificial economy simulation. This paper concludes with some conclusions about the work and gives some recommendations for future work.

2. LITERATURE
2.1 State of the art
There are many reasons why different researchers might need help to build models [3]. Beside the prediction of an outcome, the need to build a model is related also to other goals such as explaining, collecting data, getting a better insight, checking options and even for education purposes. The purpose of ABM is to help building better models, which later may be simulated to fulfill these goals. Its methodology includes mimicking of real beings or systems behavior in an artificial environment where they interact with each other. ABMs are mostly used to study a CAS system. Such systems are made of many interacting units that adapt by using their limited information from the environment [4]. Economy may be considered like a CAS system where these units may be considered as economic units, which interact with each other and adapt to new situations based on the information they gain from the economic environment. In an artificial economy, it is considered as a system where the economic units are called agents and they are simulated by using ABMs. These simulated agents are considered as autonomous and the result of their interactions can be computed [7]. Such techniques used in economic studies are known as ACE. Based on the properties that different agents in ACE share, it is suggested [8] that agents are divided in three main paradigms: simple programmed, autonomous and human-like agents. The simple programmed agents, as the name suggests, include agents programmed to follow simple rules. Autonomous agents should have a certain freedom level to explore the environment and human-like agents are the ones calibrated or set up with personal traits to make decisions in markets. There is still a need to research and create frameworks that support agents that fulfill these three agent paradigms.

There are several researches that consider ACE to simulate an economy. The first and most known of them is the project EURACE [9]. This project aimed to build an economic model of the European Union markets and it took around three years of work to make the simulation. The model was built by using

the FLAME framework and it used the best supercomputers for communication between agents. A problem related to this project is that it is not portable and it would take a lot of time to make changes for a new economic system. Several other works tried to recreate similar simulations, from whom it is depicted the Java Agent-based MacroEconomic Laboratory (JAMEL) framework [10]. The model permits the researcher to investigate the markets and the behaviors of firms. Basically, by using this framework as a tool, researchers may change the configurations to study the market from different perspectives. One disadvantage that this tool has is that it doesn't permit the user to change behaviors of agents upon the three agent paradigms. Another research work [11] that simulate economy on a smaller scale is built to experiment a closed economy. The model is programmed in the Repast Simphony framework, and it simulates only two types of autonomous agents, the person and firm agent, which interact with each other and evolve within the environment. The simulation permits the user to control different parameters such as number of agents up to few thousand and the work capacity of person agent. The model is self-regulated based on the interactions between the agents, from where it is studied the labor and goods market.

2.2 A new proposal

The simulation of an artificial economy would be possible by using ABM tools, meanwhile an economy can be considered as a CAS system. From a different point of view, CAS systems should include three important features of the units where they should be reactive, goal-oriented and planner units [12]. This means that the agents in ABM should be reactive when they perceive changes in their environment and adapt on new situations. These agents need to be goal-oriented and directed towards the achievement of their goals which may evolve during the time, and to plan for further steps to be taken.

There exist many ABM programming tools which provide their own suite of libraries and agent frameworks. These tools are based on an internal scheduler that permits them to be used as simulators. Some researchers consider the notions of ABM and MAS as synonymous while others argue that they are distinct from each other in the sense of how they are used [13]. It is possible however to build new tools based on MAS that use such internal scheduler and permit those to simulate, creating in this way new ABMs. A prominent MAS framework that fulfills such criteria and includes the features of a CAS system seems to be JaCaMo [14].

The framework of JaCaMo is a multi-agent oriented programming platform (MAOP) [15], which includes other paradigms such as agent-oriented programming (AOP), environment-oriented programming (EOP), interaction-oriented programming (IOP) and organization-oriented programming (OOP). This framework makes use of autonomous agents programmed in Jason, working in environments programmed as artifacts in CArtAgO, organized by Moise and interacting with each other using an interaction language.

Jason [16] is an AgentSpeak interpreter that supports creation of belief-desire-intention (BDI) agents and it incorporates the AOP. The structure of a Jason program is made of beliefs, goals and plans, which is very helpful to build the agent. The beliefs of Jason are considered as mental notes and are acquired either by communication or environment perception. Beliefs permit us to program rules which make possible for the agent to reason, this is quite the same with Prolog language and is part of logic programming. Goal is very fundamental and it expresses what the agent wishes to bring out as state properties. In AgentSpeak there are two types of goals known as achievement goal and test goal. Another important part is the plans which have three main parts, a triggering event which might happen from addition or deletion of belief, achievement goal or test goal, the context which checks state beliefs of the agent under which the plan is triggered, and the body of the plan which is a sequence of actions. The agents are capable to send messages to each other as part of their communication.

Common Artifact infrastructure of Agents Open environments, also known as CArtAgO [16], is a platform that permits to create several virtual environments in a MAS. It uses the Agents and Artifacts (A&A) Model and supports the EOP. This platform when used in JaCaMo permits the Jason agents to communicate and interact with each other on several environments. The idea when using CArtAgO is that an agent may be situated on different environments at the same time as a real person would do, for example: a student is part of the university environment but at the same time he is part of his family environment and he may share same environment with his friends. Moise [17] includes an organization modelling language which describes an organization by using role, group, mission, goal, social scheme and norms entities.

Prometheus [19] is a methodology for modeling intelligent systems in MAS, and many MAS developers have been using it for quite a long time. While trying to use it in JaCaMo, many researchers have found limitations and divergencies on the modeling part, as they don't cover many of its aspects [20]. Meanwhile by considering the best practices of ABM models in economics and finance, it is shown that the use of Unified Modeling Language (UML) is quite important for such models [21]. A similar modeling structure has been found in literature [22], that uses specifications of UML to help building a MAS. This paper proposes such a structure to be used altogether with the JaCaMo framework to build an artificial economy.

3. CONCEPTUAL MODEL
3.1 Model Description

To make the simulation possible, it is needed to properly describe the main activities and actors that will be part of the simulation world. As part of the considered artificial economy, it is taken into consideration a classical economy that has no regulatory institutions, such as government or other similar institutions and no tax is applied. The main economic entities taken into consideration are; the person, firm and bank.

The person is considered as a household and takes four roles in the model. The first role of the person is the consumer, the person as an economic entity has a need to consume so he may survive as a living being by maintaining a certain health. This role is very important for the simulation as it works as the key for the continuity of the simulated world. As a consumer, the person needs to have cash money to buy goods from firms. For this reason, the person needs to have a second role which is the employee role. This role is very important for the person as it is one of the main sources of income which permits him to increase his cash money. Another important role related to the money management of the person is the client role. In this role, the person acts as a client of the bank. He may deposit money when he has more than his basic needs, so he can gather them and take some extra income out of it. When needed he may withdraw money from bank, and if

there is no money deposited, he may take a loan from it and pay it back later together with some interest. The last role of

the person is the investor role, where the person may invest by buying shares from the firm or by creating a new firm.
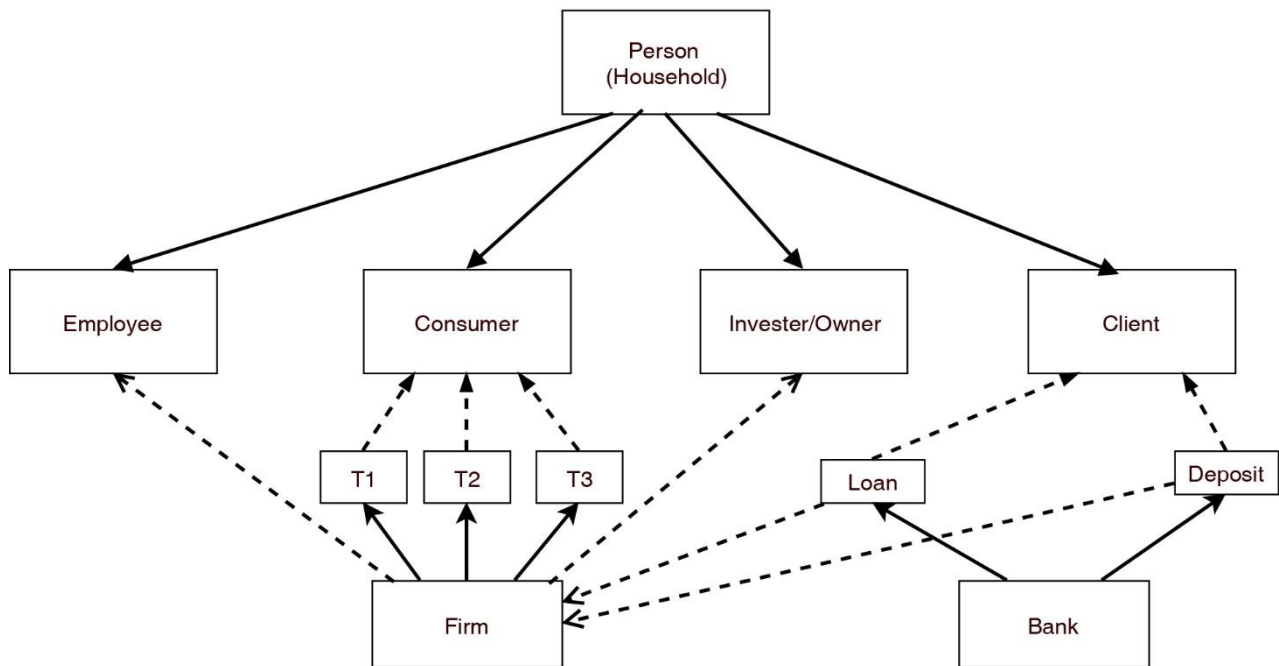

Fig. 1. Agents, roles and their relation in artificial economy

In the figure 1, it is presented the person with all his roles and the relations towards the other economic entities. The firm is a very important economic entity that fulfills the needs of the consumer. In the simulated model, it is considered that there are three types of goods to be consumed by the consume. They are divided in three types of goods named T1, T2 and T3. The T1 type is a basic good that are necessary to be consumed every day by the consumer, in this category in real terms may be considered food and other daily needs. The T2 type are the goods which are consumed less frequently by the consumers, but are necessary to them. In this type are included the necessary goods and services that persons buy several times per week or month. The last type is related to the goods which are considered as luxury and only people who have a certain wealth may consider buying them. Based on these three types of goods, each firm produces only one type of good. Based upon the demand and offer of the consumers the firms decide about the price of goods.

For the firm to be producing the above-mentioned goods, it needs to have employees and capital. Related to the employees, the firm employs person whenever it needs to produce goods and maintains them employed until it reaches a certain level of inventory, no capital money, or there is no demand for its products. The firm needs also a capital to pay the employees and other activities that it has. Its main source of capital is coming from the goods it sells, but it may have also other sources like getting new investment. The firm like the person, may have a bank client role and it can take loans or make deposits.

The bank agent is considered as a facilitator of the circulation of money in the simulation world. It sets up its own interests for loans and deposits based on the requests from the clients. It may give loans based on the money available in the bank, and may take deposits based upon the interests published by the bank on the artificial economy.

3.2 Detailed Analysis

Based on the description of the artificial economy, it is necessary to establish first the main daily activities that the agents will perform during the simulation. In the figure 2 are shown the daily activity plans for each agent. The person each day needs to update his beliefs related to his wealth, he updates information about the cash, money he has deposited or loans that he has taken from bank. In the second step the person if employed continues working on the firm. If the person is not hired than he starts the *jobsearch* activity, where he registers as a person that searches for a job and waits to get employed by a firm.

During the *workandlive* activity, the person also updates his health power which decreases automatically based on the work done. His health power updates again in the subsequent activity *consumeandlive* where it increases back again depending on what the person has consumed. In the adjustments activity plan, the person does some reasoning which will help him to make decisions, and updates his beliefs. The last activity plan is related to money decisions that the person should make to better manage his finances.
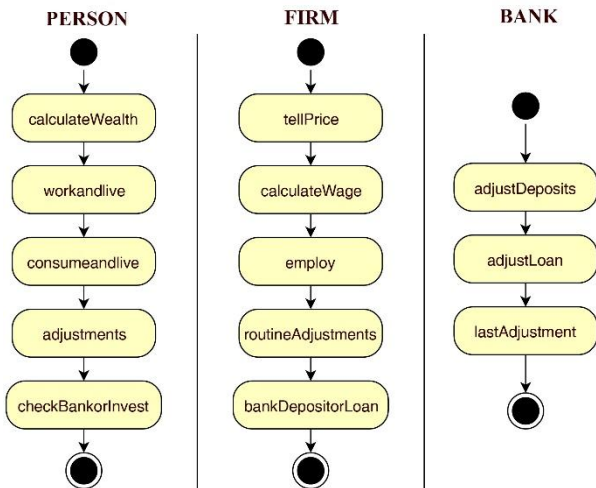
Fig. 2. Daily activity plans for each agent

The first activity that firms normally do is the announcement of the product price, this is done daily especially if the price changes frequently. Another daily activity is the calculation of wages for all working employees. The firm checks his belief to produce more, and if it is needed activates the employ plan to get more workers. If the firm doesn't need to produce or the demand is lowered based on its reasoning, then the firm may activate the *unemploy* goal plan for this case. Similar reasonings of the firm agent are done in the *routineAdjustments* plan, where the firm may ask for new investments or to follow some new strategies in the market. The last activity plan to be activated daily on the firm is the

bankDepositorLoan, where the firm agent takes decision to deposit or to take a loan from bank based on its monetary needs.

The bank daily activity plans are related to deposit and loan account checking, and some important reasonings to change the interest rates based on money demand and offer.

The agents need to apply these plans based on the information they share with each other inside their own environments. As part of the interactions that these agents have with each other, it is necessary to think of markets. In this respect the markets will be the place where the agents will communicate by sharing the needed information about their beliefs. The markets thought in this work are related to the roles of person agent and are divided as labor, stock, goods and deposit-loan market. The very first three markets which are related to firm agent, are included in an artifact named *FirmEnv*, and serves as the environment of person-firm interactions. Meanwhile another artifact named *BankEnv* keeps information on deposit-loan market and serves as the environment of bank interactions with the other agents.

In figure 3, it is shown a sequence diagram related to money management of the person agent. The person agent checks his wealth by sending a request to *BankEnv* artifact. The artifact communicates with the respective bank object that keeps the data about deposits and loans, and checks the accounts with the id of the person. By using the methods *getDepsTotal*() and *getLoansTotal*(), the bank gathers all amounts of deposits and loans related to the person and calculates their total. The difference of these totals is the reply that the artifact sends back to the person agent.
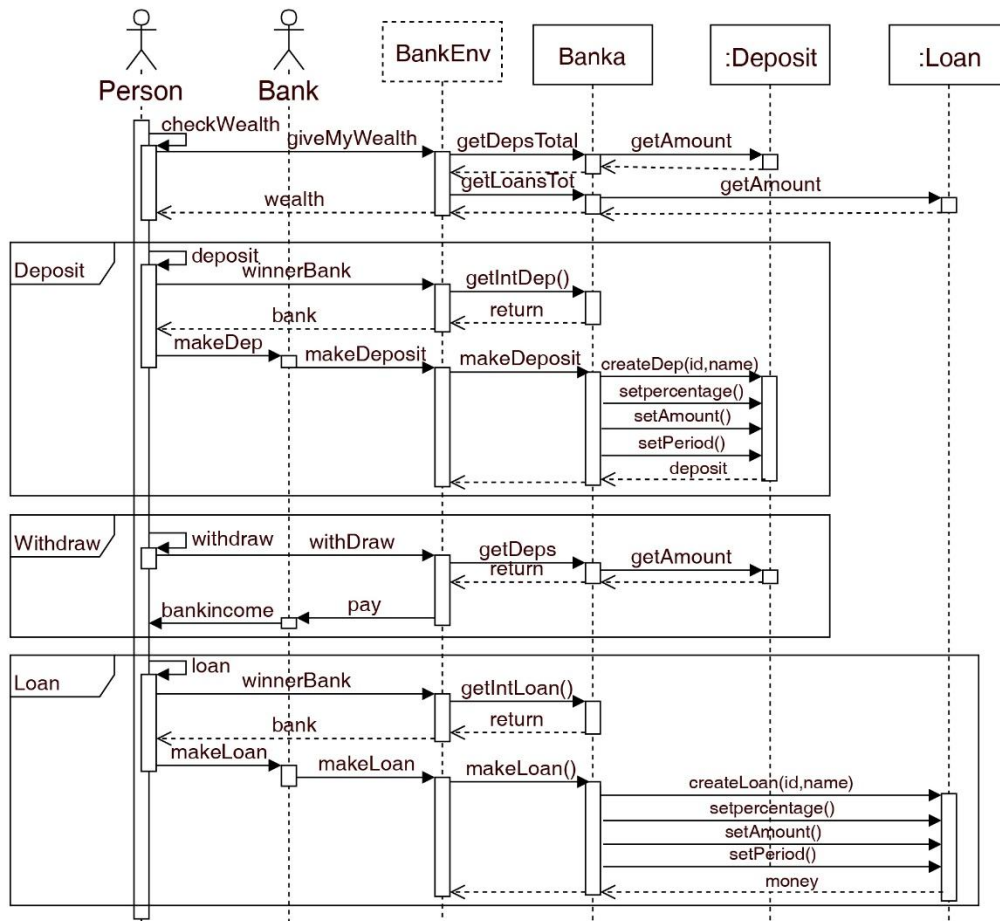

Fig. 3. Money Management Sequence Diagram

The deposit activity initiates from the person agent whenever he fulfills the condition of having more cash than he needs to consume daily for a certain period. The agent asks the *BankEnv* artifact to find the best bank based on the deposit interest and its relative location from the agent itself. After finding the bank, the person agent sends a message to bank agent to make a deposit and the data are transmitted to the respective bank object where a new deposit object is created. The respective beliefs of cash and money are updated in both person and bank agents. A similar approach is followed also for the loan activities.

The withdraw activity initiates whenever the person agent is in need for money. In this case, the person sends a message to the *BankEnv* artifact where it asks to withdraw money from its deposits created. The deposits found are sent back to the respective bank agent to process with the withdraw payment. At this moment the bank agent updates its beliefs about the deposit changes and sends to the person agents the cash which was asked by him.

In the figure 4 is shown the sequence diagram of labor market activities that the person and firm agents perform related to employment. As part of the person agent, it is considered that sends his data to the *FirmEnv* artifact to be added to the list of *AskJobs* objects. Whenever a firm agent sends an employ request to *FirmEnv* artifact, it checks for the best job asker by using a reasoning based on experience and location, and it sends the information back to firm agent to hire the person agent. On this moment the person id is added to firm agent belief base and a hired message is sent to the person agent.
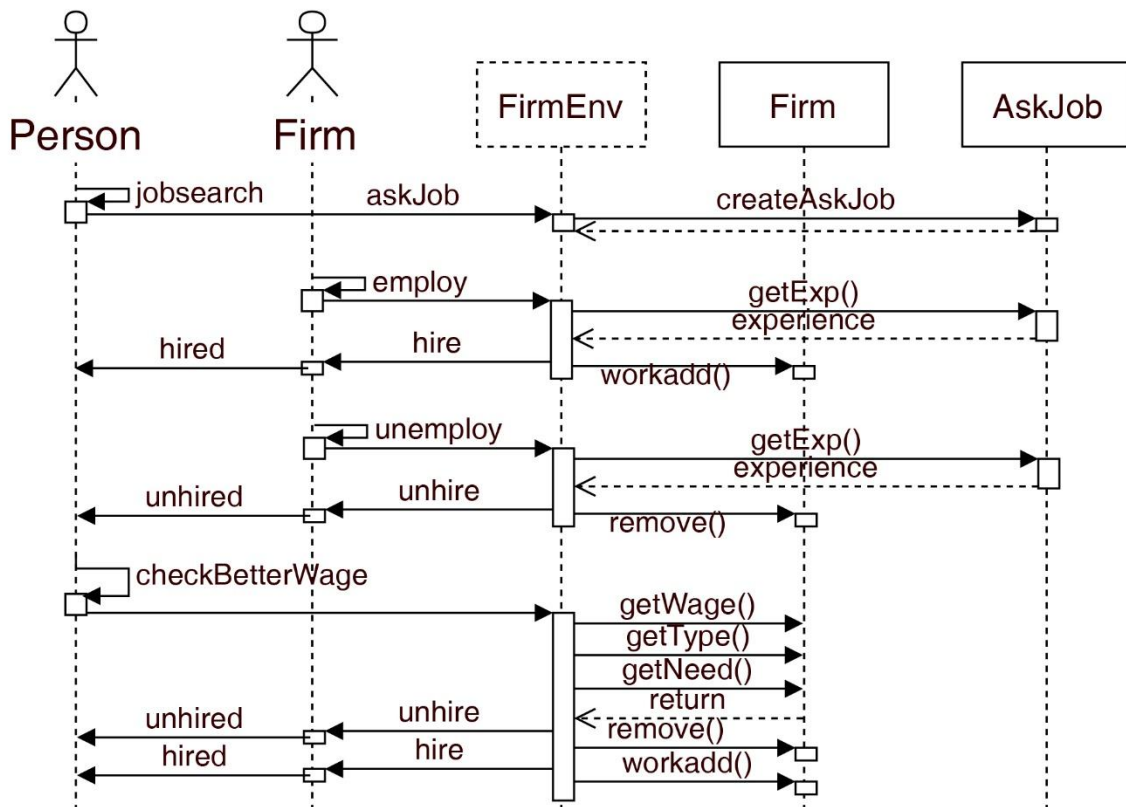

Fig. 4. Sequence Diagram of employment

Occasionally, the person agent may check for a better wage in the market. In this case a request is made to the *FirmEnv* artifact to do the reasoning about the possibility to change the workplace to a new job. If the conditions are fulfilled and a better offer is found the person agent gets *unhired* from the respective firm and gets hired on a new firm. As part of the relations between person and firm agents, there are also other processes that include the interactions of person and firm agents based on the stock and goods markets.

4. IMPLEMENTATION
In the previous section the conceptual model is presented with details of the artificial economy simulation operations. This section presents the implemented application based on JaCaMo framework which is explained in the proposal section of the literature review. The implementation model used in this work makes use of its two main components: Jason and CArtAgO.

In figure 5 is presented the implemented logic of the artificial economy simulation. Its main idea is that the coordinator agent is the main controller of the simulation, for this reason on the moment it is created by the environment parser of JaCaMo the coordinator initializes a configuration panel based on *GUIArtifact* which awaits inputs from the researcher. This *GUIArtifact* is part of CArtAgo and provides the necessary graphical user interface (GUI) to be associated with the artifact. In this case the configuration panel after receiving input from the researcher, it triggers new events on the coordinator by changing its beliefs. The researcher may choose the number of persons, firms, and banks as well, he may change the initial working capacity of workers for different types of firms as well as the initial labor cost. Based on these input data, the user may start the simulation by clicking on start button which sends a signal to coordinator agent to setup the initial conditions and create the agents in the artificial economy.
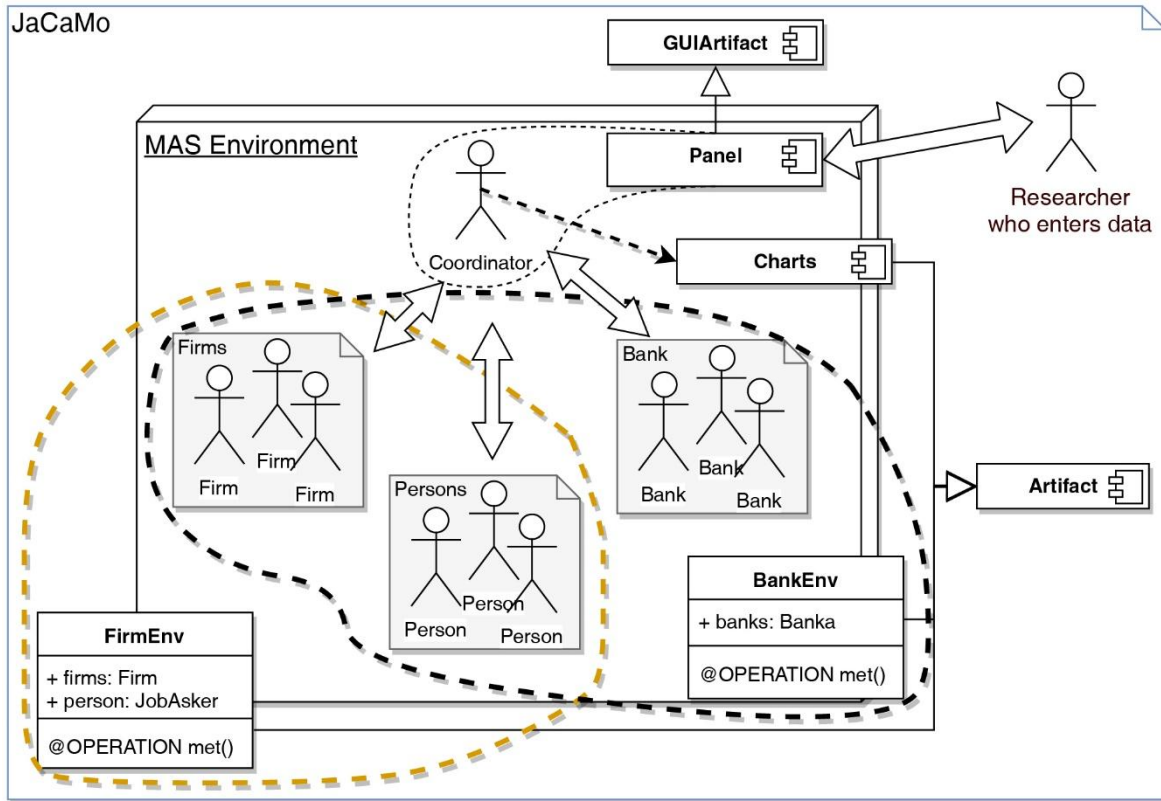
Fig. 5. The Implemented Infrastructure of Artificial Economy

In Listing 1, it is shown a part of the coordinator code which creates the *Panel* artifact, by calling its class in Jason code. The artifact uses a signal method to send a triggering event by belief addition. On the moment that the Panel artifact sends as belief the command create with the number of person, firm and bank as parameters, the plan *create(P,B,F)* is triggered and in a sequential way the achievement goals execute the respective plans for the creation of these agents. In the plan to create persons, line 7 up to 12, it is used a for loop that executes a sequence of activities, starting from the testing of person list, creating the name of person agent and adding it to the person list and in the end to create the person agent by following the rules on *person.asl* Jason file. By using this technique, there are created several autonomous agents that are added to the simulation.

```
1  +!initiate
2  <-  makeArtifact("Panel","artificial_economy.Panel",[],Id);
3         focus(Id) .
4         +create(P,B,F)<-
5         !create_person(P);!create_bank(B); !create_firm(F);
6         !start .
7         +!create_person(N) <-
8           for(.range(I,1,N)){
9              ?prs(X,K);.concat("person_",K+1,CN);
10             .concat([CN],X,Z);  -+prs(Z,K+1);
11              .create_agent(CN,"person.asl",
12              [agentArchClass("jaca.CAgentArch")]);
13        } .
```

Listing 1. Part of coordinator.asl code to make artifact and create agents

After the researcher has clicked the button start on the Panel artifact, it sends a signal to the coordinator to add the agents as shown in the listing 1. After achieving the goals to create agents, the achievement goal start is executed. In the listing 2,

the plan of this goal is shown. The plan adds a new belief named iteration with value 1 and then executes the iterator plan which can be considered as the ticking clock of this simulation. For each period, the plan iteratively executes the plan *dailyinit* and then itself again. This plan sends goal achievement messages to the agents of the simulation to update their ticking clock and to execute their routine plans. In this way, the coordinator agent is used as an internal scheduler that simulates a ticking clock and the daily actions of the agents. The routine plans of agents together with their beliefs and reasonings are programmed in Jason code as described in detailed analysis section. Communication and reporting of these agents from and to the coordinator is shown in figure 5.

```
30  +!start: true <- +iteration(1);        !iterator.
31  +!updateTick<-?tick(N);-+tick(N+1); updateTime(N+1).
32  +!iterator:iteration(N)&N=1<- wait(500);
33       !dailyinit;  !iterator.
34  @p1[atomic]
35  +!dailyinit : tick(N) <- ?prs(P,_);
36       .send(P,achieve,[updateTick(N),routine]);
37       ?t1(A,_);?t2(B,_);?t3(C,_);   .concat(A,B,C,D);
38       .send(D,achieve,[updateTick(N),routine]);
39       ?bnk(J,K);
40       .send(J,achieve,[updateTick(N),routine]);
41       !updateTick.
```

Listing 2. Part of coordinator.asl code to schedule and call daily activities from agents.

The relation of agents and artifacts is important in this work. The artifacts are made by the respective agent plan and are shared through them by simulating economic markets. For example, the artifact *BankEnv* is an artifact created when the first bank agent was created. In this artifact are stored the important data about the deposit and loan accounts of bank

clients. The bank clients, namely the person and firm agents, need to ask about their deposits or loans in banks, they may send a request to this artifact to update their beliefs. These artifacts may be perceived as facilitators of reasoning process of the agents as they provide information about the environment. *FirmEnv* is another important artifact like *BankEnv* which is used in the simulation. This artifact is shared between the person and firm agent, and acts as an environment of three important markets: the labor market, the goods market and stock market. Based on the role that the person agent shows at a certain moment, it accesses the proper operations from the artifact.

Listing 3 shows two important operations, namely the *askJob*() and *employ*() methods. Based on the description of figure 4, the person agent adds a belief of *jobsearch* which activates a plan that communicates with the *FirmEnv* artifact to call the operation *askJob*() with the parameters of the person agent. This operation stores the data of the agent in a list of *jobaskers*. Later when the firm adds the employ belief, it activates the plan to call the operation employ from the *FirmEnv* artifact with the name of the firm agent. At this stage, the artifact does the reasoning based on the list and chooses the best possible candidate to get employed by the firm, and it sends a hire signal to the firm based on the wage the firm offers. be numbered (e.g., "Table 1" or "Figure 2"), please note that the word for Table and Figure are spelled out. Figure's captions should be centered beneath the image or picture, and Table captions should be centered above the table body.

```
1  public class FirmEnv extends Artifact {
-- ...
5      @OPERATION void askJob(...){
6          jobaskers.add(new AskJob(...));
7      }
-- ...
23     @OPERATION void employ(String firm){
24         if(jobaskers.size()>0){
           AskJob personToHire=jobaskers.get(0);
--         ...
32         for(int i=0;i<jobaskers.size();i++){ ... }
--         ...
49         jobaskers.remove(personToHire);
50         signal("hire",firm,personToHire.getPerson());
51         personToHire.setWage(wage);
52         for(Firm x:firms) if(x.getFirm().equals(firm))
53             { x.getWorkers().add(personToHire); }
54         } else {  signal("nohire",firm); }
55     }
--- ...
205 }
```

Listing 3. Parts of FirmEnv.java code with operations of askJob() and employ().

These artifacts create and modify agent objects which keep states of the simulated agents. These objects are used to help agents to make better reasoning processes and to make use of all benefits that OOP may offer. The related objects created in *FirmEnv* artifact are the Firm object which represents important data about the firm agent, and *AskJob* object which represents the person agent that asks for a job. In the *BankEnv* artifact, the object Bank is the representation of bank agent and it uses the Deposit and Loan object to store the deposits and loans that agents make on the bank agent.

Another important feature implemented in this simulation is the reporting mechanism it uses. Every agent at the end of the day reports to the coordinator its data. These collected data are sent by the coordinator agent to its Chart artifact. This artifact makes use of the open source JFreeChart library, and it constructs a live dynamic graph of the agent states, providing in this manner a live feedback from the artificial economy simulation. In a similar way, these data are written on data files which can be used later by the researcher for further studies.

5. CONCLUSION AND FUTURE WORK

The failure of traditional methods to predict and prevent several economic crises has pushed many researchers to find other methods to study economy. One of these methods is the use of agent-based methods to create artificial economy models. Based on the investigation done in the state of the art of this paper, it is concluded that there is still a need to create better frameworks that support many of the features that this research community needs.

The paper considers a new proposal to use a prominent MAS framework, known as JaCaMo, for creating an artificial economy based on ABM principles. This framework combines the use of different programming paradigms to act as a MAOP, and it includes the Jason as a language to program BDI agents, CArtAgO as a language to program environments as artifacts and Moise as a language for creating organization. Based on the best practices of agent-based models in economy, the paper firstly provides a general description as a requirement for the artificial economy. Then it continues with a detailed analysis of the agent processes. Firstly, the main daily activities of these agents are described, followed by some sequence diagrams to show the interactions between the agents. This analysis methodology together with some UML design diagrams helps the developer to be concise on the details that need to be implemented later for the simulation. The implementation section describes in detail the implementation of the artificial economy infrastructure. It should be noted that the role of the coordinator agent presented in this paper is very important. Whenever the simulation starts only the coordinator agent is activated, and after it initializes the Panel artifact, it awaits user input to create the other agents of the simulation. This agent has an internal scheduler that permits the simulation to be run and it provides the needed reporting mechanisms such as graphs and data files that are used by the researcher.

The model used in this work is based on a simple classical economy that has people, three types of firms and banks. No other institution, such as government or other regulatory institutions were taken into consideration. In the future, it is intended to scale-up this model to include other institutions and agents, as well as to add other features to the existing agents. The idea is to create a fully operational artificial economy that mimics real economies.

Based on the advantages that JaCaMo framework provides as a MAOP platform, it is possible to consider other new features by considering the organization-oriented programming and the agent-artifact communication. The organization-oriented programming might be a good research point that future ABM researchers may consider, as it makes possible to create programmed social structures. Similarly, the capability of the agent-artifact communication offered by CArtAgO might be explored for the inclusion of different machine learning algorithms to help in the decision-making process of the agents.

6. REFERENCES

[1] Bouchaud, J.-P., Economics needs a scientific revolution, Nature, Vol. 455, pp. 1181, 2008.

[2] Izquierdo, L. R., Izquierdo S. S., Artificial Economics: What, Why and How, Lecture Notes in Management and Industrial Engineering, 2015.

[3] Epstein, J. M., Why model?, Journal of Artificial Societies and Social Simulation, Vol. 11, Iss. 4, pp. 12, 2008.

[4] Holland, J. H., Studying Complex Adaptive Systems, Journal of Systems Science and Complexity, Vol. 19, Iss. 1, pp. 1-8, 2006.

[5] Held, F., Wilkinson, I., Computer Simulation and Agent-Based Models as a Research Method, in Collaborative Research Design, Singapore, Springer, 2018, pp. 377-398.

[6] Hakrama, I., Artificial Economy and the usage of ACE, in Information Systems and Technology Innovations: Projecting trends to a New Economy, Tirana, Albania, 2014.

[7] Richiardi, M. G., Agent-based computational economics: a short introduction, The Knowledge Engineering Review, Vol. 27, Iss. 2, pp. 137-149, 2012.

[8] Chen, S.-H., Varieties of agents in agent-based computational economics: A historical and an interdisciplinary perspective, Journal of Economic Dynamics and Control, Vol. 36, Iss. 1, pp. 1-25, 2012.

[9] Deissenberg, C., Hoog, S. v. d., Dawid, H., EURACE: A massively parallel agent-based model of the European economy, Applied Mathematics and Computation, Iss. 204, pp. 541-552, 2008.

[10] Seppecher, P., Salle, I., Lang, D., Is the Market Really a Good Teacher?: Market selection, collective adaptation and financial instability., in 20th Conference of the Research Network Macroeconomics and Macroeconomic Policies - Towards Pluralism in Macroeconomics?, Berlin, Germany, 2016.

[11] Hakrama, I., Kraja, I., The self-regulated model of a closed economy: An agent-based simulation model for experimental purposes, in Information Systems and Technology Innovations: New Paradigm for a Smarter Economy, Tirana, Albania, 2016.

[12] Gevel, A. J. W. v. d., Noussair, C. N., The Nexus between Artificial Intelligence and Economics, Berlin Heidelberg: Springer-Verlag, 2013, pp. 82.

[13] Niazi, M., Hussain, A., Agent-based computing from multi-agent systems to agent-based Models: a visual survey, Scientometrics, Vol. 89, Iss. 479, 2011.

[14] Hakrama, I., Frasheri, N., Modeling an artificial economy with JaCaMo, in 10th Annual South-East European Doctoral Student Conference, Thessaloniki, Greece, 2015.

[15] Boissier, O., Hübner, J. F., Ricci, A., The JaCaMo Framework, in Social Coordination Frameworks for Social Technical Systems, Switzerland, Springer, 2016, pp. 125-151.

[16] Bordini, R. H., Hübner, J. F., Wooldridge, M., Programming Multi-Agent Systems in AgentSpeak using Jason, John Wiley & Sons, 2007.

[17] Ricci, A., Piunti, M., Viroli, M., Omicini, A., Environment programming in CArtAgO, Multi-Agent Programming: Languages, Platforms and Applications, Vol. 2, pp. 259-288, 2009.

[18] Hubner, J. F., Sichman, J. S., Boissier, O., Developing organised multiagent systems using the MOISE+ model: programming issues at the system and agent levels, Int. J. of Agent-Oriented Software Engineering, Vol. 1, Iss. 3/4, pp. 370-395, 2007.

[19] Padgham, L., Winikoff, M., Prometheus: A Methodology for Developing Intelligent Agents, Agent-Oriented Software Engineering III, Lecture Notes in Computer Science, Vol. 2585, pp. 174-185, 2003.

[20] Freitas, A., Cardoso, R. C., Vieira, R., Bordini, R. H., Limitations and Divergences in Approaches for Agent-Oriented Modelling and Programming, in Workshop on Engineering Multi-Agent Systems (EMAS-16), Singapore, 2016.

[21] Vermeir, A., Bersini, H., Best practices in programming agent-based models in economics and finance, Advances in Artificial Economics, pp. 57-68, 2015.

[22] Hakrama, I., Tabaku, R., Modelling and Implementation of a virtual warehouse through Jason and RMI, in 4th International Conference on Advanced Technology & Sciences (ICAT'Rome), Rome, Italy, 2016.

[23]